



## PATENT ABSTRACTS OF JAPAN

(11) Publication number: **02019938 A**(43) Date of publication of application: **23.01.90**

(51) Int. Cl

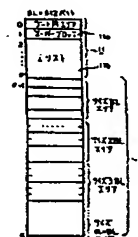
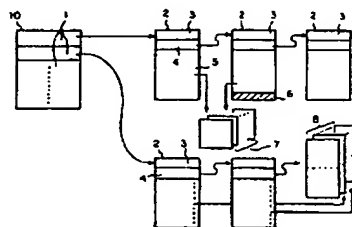
**G06F 12/00**(21) Application number: **63168981**(71) Applicant: **MITSUBISHI ELECTRIC CORP**(22) Date of filing: **08.07.88**(72) Inventor: **MIKAMI KAZUYOSHI**(54) **FILE CONTROL SYSTEM**

(57) Abstract:

**PURPOSE:** To control files at a high speed by using a continuous block control table to control the block length prescribed at initialization of a file and the number of blocks or preparing each idle block header group.

**CONSTITUTION:** A continuous block control table 10 stores the idle block list headers 1 for each of sizes BL, 2BL...  $m \times BL$  arranged in order of larger ones. At initialization of a file, a user area 12 is divided every integer multiple of the minimum block size BL and continuously. Thus the list structures are initialized for each size  $m \times BL$ . When a host machine requests a writing action of the data having its length (x), a file system returns the address of an idle block. In other words, the size of a desired idle block is calculated as  $(x/BL) + 1$ . Thus a single access suffices to the files allocated to blocks and therefore the access speed is increased.

COPYRIGHT: (C)1990,JPO&amp;Japio



Best Available Copy

## ⑫ 公開特許公報(A) 平2-19938

⑬ Int. Cl.<sup>5</sup>  
G 06 F 12/00識別記号 庁内整理番号  
3 0 1 B 8944-5B

⑭ 公開 平成2年(1990)1月23日

審査請求 未請求 請求項の数 1 (全6頁)

⑮ 発明の名称 ファイル管理システム

⑯ 特 願 昭63-168981

⑰ 出 願 昭63(1988)7月8日

⑱ 発 明 者 三 上 和 敬 兵庫県神戸市兵庫区和田崎町1丁目1番2号 三菱電機株式会社制御製作所内

⑲ 出 願 人 三菱電機株式会社 東京都千代田区丸の内2丁目2番3号

⑳ 代 理 人 弁理士 田澤 博昭 外2名

## 明 細 書

## 1. 発明の名称

ファイル管理システム

## 2. 特許請求の範囲

連続ブロック管理表内に格納された最小ブロック長の整数倍の各サイズごとの空きブロックリストヘッダ群あるいは個々の空きブロックリストヘッダ群と、アドレスの個数を格納するエリアと次のリストを示すポインタを格納するエリアおよび最小ブロックサイズの整数倍ごとで連続エリアで分割した複数の空きブロックのアドレスを格納するエリアとを有し、ファイルの初期化動作時に上記空きブロックのアドレスを格納するエリアに連続して空きブロックを保持しかつホストマシンのアクセス時に上記空きブロックリストヘッダ群の示す番号を上記アドレスを格納するエリアにセットしてそのアドレスをホストマシンに通知するリストとを備えたファイル管理システム。

## 3. 発明の詳細な説明

〔産業上の利用分野〕

この発明は、情報処理装置のファイル管理システムに関するものである。

〔従来の技術〕

第5図は例えば共立出版、石田晴久著「UNIX」第7章「UNIXファイルシステムの内部構成」に示された従来の空きブロックのファイル管理システムの説明図であり、図において、1は空きブロック群をリスト構造で管理するための先頭リストを示す空きブロックリストヘッダ、2は最大N(N=50)個の空きブロックのアドレスを保持するためのリスト、3はリスト2中に保持されているアドレスの個数Fを格納するエリア、4は次のリストを示すポインタを格納するエリア(次のリストが存在しないときは終端マークとしてφを格納)、5は空きブロックのアドレスを格納するエリア、6は使用中のブロックが不要になったとき、すなわち、新しい空きブロックができると、そのアドレスを記入できるN-F個のエリア、7はエリアによって示される空きブロックである。

次に動作について説明する。UNIXシステムの

ファイルの初期化、すなわち、フォーマット時において、ファイルシステムは第6図に示すような管理用エリア11(スーパーブロック11a、1ノードの集まりである1リスト11bからなる)を設けて、ユーザエリア12を全て空きブロック(固定サイズ512バイト)として、第5図に示すリスト構造を生成して空きブロックを管理する。このスーパーブロック11aには、第5図の空きブロックリストヘッダ1が格納され、ユーザエリア12の各ブロックは空きブロック7に対応している。

次にホストマシンがファイル書き込みに当たって、ブロック割当てを要求した場合の動作について説明する。ホストマシンがデータ長 $x$ バイトのデータの書き込みを要求すると、ファイルシステムは第7図のフローチャートにしたがって、必要ブロック数 $K$ に対応する空きブロックアドレス( $K$ 個)を返すことで、そのブロック割当てを行っている。

すなわち、ステップST1で必要とする空きブロック数 $K$ (整数)を、 $K - \frac{x}{L} + 1$ ( $L$ はブロック長、

$F$ は第1リストにおいて、管理している空きブロック数)として算出し、空きブロックリストヘッダ1の示すリスト2を第1リストにセットする。

次いで、ステップST2でホストマシンが第1リストにアクセスして、ステップST3で $F - K > 0$ の判断を行い、 $F - K > 0$ のとき、ステップST4に移行する。

このステップST4において、リスト2中に保持されているアドレスの個数 $F$ を格納するエリア3の値の更新を行い、 $F$ を $F - K$ とし、ステップST5で空きブロックのアドレスを格納するエリア5のボトム方向から $K$ 個の空きブロックアドレスをホストマシンに通知する。

また、ステップST3において、 $F - K > 0$ でない場合には、ステップST6に移行し、エリア5のボトム方向から $F$ 個の空きブロックアドレスをホストマシンに通知し、ステップST7で第1リストのエリア4のポインタを示すリストを次の第1リストとし、ステップST8で必要ブロック数 $K$ の値を更新して $K$ を $K - F$ とし、ステップST2に戻る。

#### 〔発明が解決しようとする課題〕

従来のファイル管理システムは以上のように構成されているので、必要ブロック数 $K$ 個は確保されるものの、そのブロックアドレスが連続アドレスになっていないため、データの書き込み時に $K$ 回のアクセスを必要とし、逆にこのファイル管理システムで書き込まれたデータを読み込むためには、 $K$ 回のアドレスを必要とするなど、高速処理に向かない問題点があった。

この発明は上記のような問題点を解消するためになされたもので、1回のアクセスで書き込みまたは読み込みが可能のように連続した空きブロックをホストマシンに返すことができ、高速処理が可能なファイル管理システムを得ることを目的とする。

#### 〔課題を解決するための手段〕

この発明によるファイル管理システムは、ファイルの初期化時にファイルシステムで規定されたブロック長とその個数を管理する連続ブロック管理表を付加するかあるいは個々の空きブロックヘ

ッダ群を設けるようにしたものである。

#### 〔作用〕

この発明における連続ブロック管理表または空きブロックヘッダ群はファイルの初期化時にホストマシンによりアクセスされ、アドレスの連続性を保持しつつ空きブロックをブリフォーミングし、その所定のブロック長を満たすブロックアドレスをホストマシンに返すものである。

#### 〔実施例〕

以下、この発明の一実施例を図について説明する。第1図において、1はサイズ( $m \times BL$ )の空きブロック群をリスト構造で管理するための先頭リストを示す空きブロックリストヘッダ、2は最大 $N$ 個の空きブロック(サイズ $m \times BL$ )のアドレスを保持するためのリスト、3はリスト2の中に保持されているアドレスの個数 $F$ を格納するエリア、4は次のリストを示すポインタを格納するエリア(次のリストが存在しないときは終端マークとして $\phi$ を格納)、5はサイズ( $m \times BL$ )の空きブロックのアドレスを格納するエリア、6は使用

中のブロックが不要になったとき、すなわち、新しいサイズ( $m \times BL$ )の空きブロックができるとそのアドレスを記入できるエリア、7はサイズ $BL$ の空きブロック、同様に8はサイズ $2BL$ の空きブロックである。

また、10はこの発明によって付加されたもので、最小ブロック長を $BL$ とすると、上から順にサイズ $BL, 2BL, 3BL, \dots$ で第 $m$ 番目では $m \times BL$ の各サイズごとの空きブロックリストヘッダ1を格納する連続ブロック管理表である。

なお、第2図の11は管理用エリア、11aはスーパーブロック、11bは1リスト、12はユーザエリアである。

次に動作について説明する。ファイルの初期化動作では、第2図に示すユーザエリア12を均一サイズでなく、最小ブロックサイズ $BL$ の整数倍ごとにかつ連続エリアに分割してサイズ( $m \times BL$ )ごとのリスト構造として初期化するものである。

ここで、各サイズごとにどれだけの個数をブリフォーミングするかの情報はブリフォーマ

ストにおいて管理しているサイズ $m \times BL$ の空きブロック数が $F-1 > 0$ か否かを判断し、 $F-1 > 0$ であれば、ステップST15に移る。

ステップST15において、エリア3に格納されている空きブロック数 $F$ の値を更新し、ステップST16でエリア5のボトム方向からサイズ $K \times BL$ の1個の空きブロックのアドレスをホストマシンに通知する。

また、ステップST14において、第1リストにおいて管理しているサイズ $m \times BL$ の空きブロック数が $F-1 > 0$ でなければ、ステップST17に移り、第1リストのエリア4に格納されているポインタの示すリストを次の第1リストとする。

このようなブロック割当てでは、ファイルの書き込みまたは読み込みは第1図のアクセスで行える。

なお、上記実施例では、各サイズごとの空きブロックリストヘッダ群とまとめて格納する連続ブロック管理表10を設けたものを示したが、このような連続ブロック管理表10を設けず、リスト

ティングのパラメータとして与え、第1図における空きブロックリストヘッダ1のリスト2上のエリア3の値の合計値に等しくなる。

次にホストマシンがファイル書き込みに当たってブロック割当てを要求した場合の動作について説明する。ホストマシンがデータ長 $x$ バイトのデータの書き込みを要求すると、ファイルシステムは第3図のフローチャートにしたがって1個の空きブロック(サイズ $m \times BL$ )のアドレスを返すことで、そのブロック割当てを行う。

すなわち、ステップST11において、必要とする空きブロック $K$ のサイズの算出を $(\frac{x}{BL} + 1)$ として算出する。ここで、 $x$ はデータ長、 $BL$ は最小ブロックサイズである。

次にステップST12で、連続ブロック管理表10の第 $K$ 番目のブロックサイズ $K \times BL$ の空きブロックリストヘッダ1の示すリスト2を第1リストにセットする。

次いで、ステップST13でホストマシンが第1リストにアクセスして、ステップST14で第1リ

スト構造、すなわち、第4図に示すように、第1ノード13にサイズ1の空きブロックリストヘッダ群、第2ノード14にサイズ2の空きブロックリストヘッダ群、第3ノード15にサイズ3の空きブロックリストヘッダ群、…というようにデータ配置を行い、各サイズごとの空きブロックリストヘッダ群を管理しても上記実施例と同様の効果を奏する。

#### [ 発明の効果 ]

以上のように、この発明によれば、ファイルの初期化時に最小ブロックサイズの整数倍サイズごとに空きブロックリストヘッダを連続ブロック管理表で格納するか、あるいはノードごとに最小ブロックサイズの整数倍サイズごとの空きブロックリスト群を管理するように構成したので、ブロック割当てされたファイルのアクセスは1回で行えるため、リアルタイム処理などの高速アクセスを実現できる効果がある。

#### 4. 図面の簡単な説明

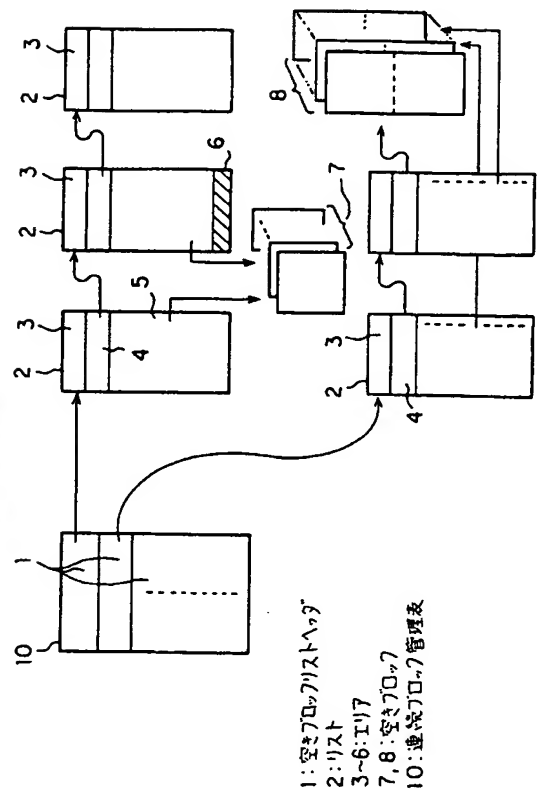
第1図はこの発明の一実施例によるファイル管

理システムの説明図、第2図は同上実施例におけるファイルの初期化を説明するためのディスクの内部構成図、第3図は同上実施例におけるファイル書き込み時のブロック割当ての動作の流れを示すフローチャート、第4図はこの発明の他の実施例によるファイル管理システムにおけるリスト構造で各サイズごとの空きブロックリストヘッダ群の説明図、第5図は従来のファイル管理システムの説明図、第6図は従来のファイル管理システムにおけるファイルの初期化を説明するためのディスクの内部構成図、第7図は従来のファイル管理システムにおけるファイル書き込み時のブロック割当ての動作の流れを示すフローチャートである。

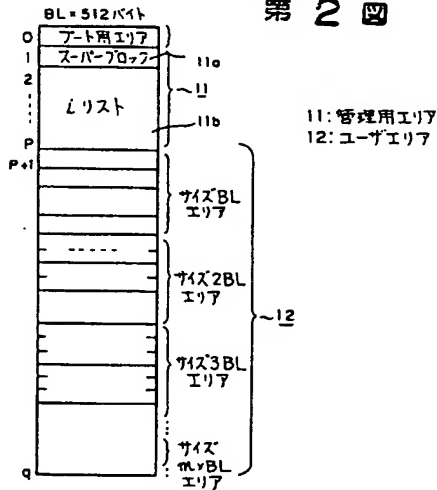
1は空きブロックリストヘッダ、2はリスト、3～6はエリア、7、8は空きブロック、10は連続ブロック管理表、13は第1ノード、14は第2ノード、15は第3ノード。

なお、図中、同一符号は同一、又は相当部分を示す。

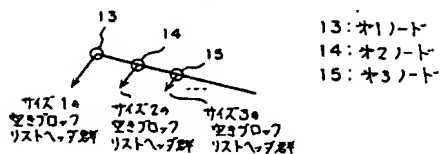
第1図



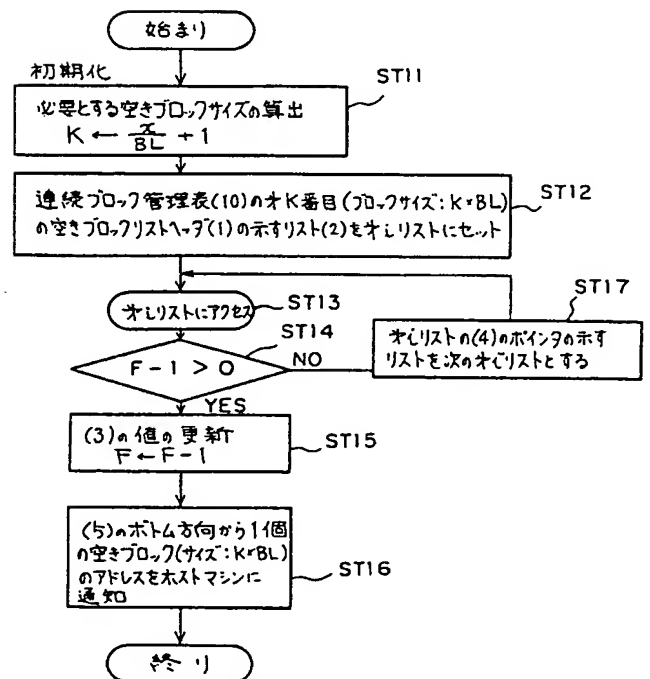
第2図



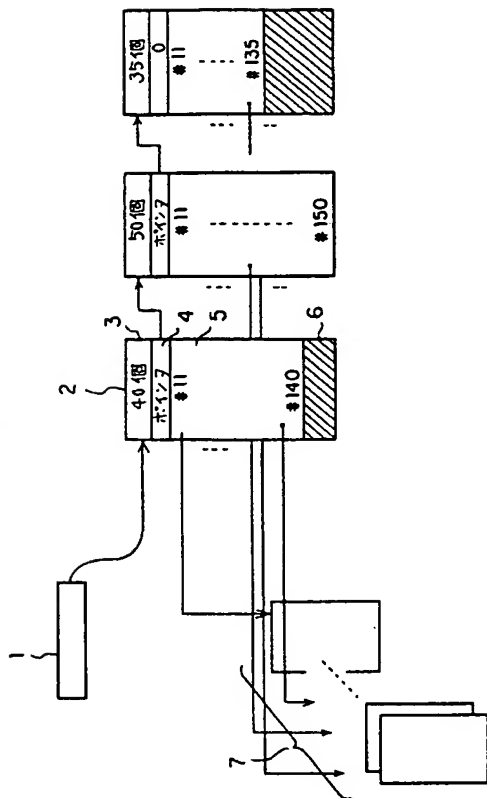
第4図



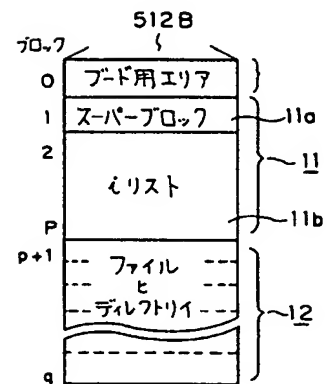
第3図



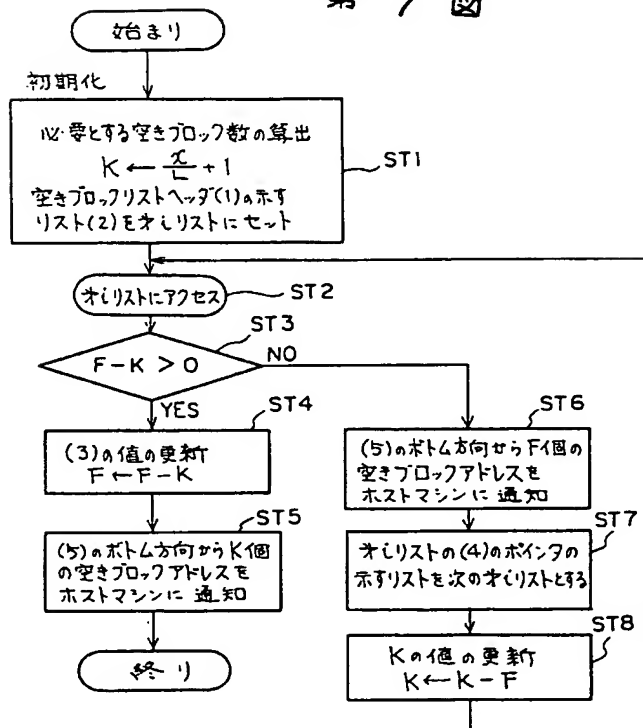
第 5 図



第 6 図



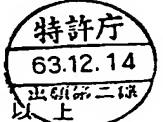
第 7 図



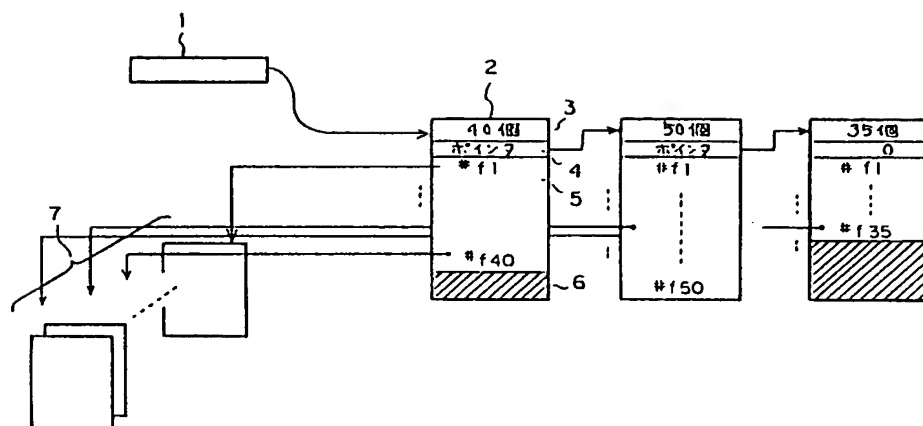
手続補正書 (自発)  
63.12.14  
昭和 年 月 日

特許庁長官殿

1. 事件の表示 特願昭 63-168981号
2. 発明の名称  
ファイル管理システム
3. 補正をする者  
事件との関係 特許出願人  
住 所 東京都千代田区丸の内二丁目2番3号  
名 称 (601)三菱電機株式会社  
代表者 志 岐 守 哉
4. 代 理 人 郵便番号 105  
住 所 東京都港区西新橋1丁目4番10号  
第3森ビル3階  
氏 名 (6647)弁理士 田 澤 博 昭  
電話 03(591)5095番
5. 補正の対象 図 面
6. 補正の内容 別紙の通り第5図を補正する。
7. 添付書類の目録 補正後の第5図を記載した書面 1通



第 5 図



**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

### **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☒ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**